

Many-one reductions between search problems

Arno Pauly

Computer Laboratory
University of Cambridge, United Kingdom
Arno.Pauly@cl.cam.ac.uk

Many-one reductions between search problems (i.e. multi-valued functions) play a crucial part in both algorithmic game theory (via classes such as PLS or PPAD) and the study of incomputability in analysis. While the formal setting differs significantly, the present papers offers a unifying approach in terms of category theory that allows to deduce that any degree structure arising from such reducibilities is a distributive lattice. Moreover, it is a Kleene-algebra, which allows to consider *wtt*-degrees, too. We discuss some specific examples and study degree-theoretic properties that do depend on the specific reducibility.

Contents

1	Introduction	1
1.1	Overview	2
1.2	Preceding work	2
1.3	A first example	3
2	A generic definition of many-one reductions	4
2.1	The category-theoretic foundations	4
2.2	The lattice of many-one degrees	8
2.3	The Kleene-algebra of many-one degrees	14
3	Standard Examples	16
3.1	Computable many-one reductions (Type 1)	16
3.2	Polynomial-time many reductions (Type 1)	17
3.3	(Continuous) Weihrauch-reductions	20
4	Non-standard examples	20
4.1	Parameterized Search Problems	20
4.2	Medvedev-reducibility	21

1 Introduction

Both computability theory and complexity theory traditionally focus on **decision problems**, an observation that especially holds for degree-theoretic considerations. In computability theory this also covers **functions** by identifying them with their graph. Standard treatments of complexity theory (e.g. [30]) often include a chapter on **optimization problems**, which can be conceived of as a special kind of **search problem** (or multi-valued function): Any instance of a problem specifies some set of solutions, one of which has to be produced.

The present paper is meant to instigate a systematic investigation of search problems under many-one reductions. There are two main reasons for this endeavour:

First, search problems induce a nicer degree structure than decision problems, in particular, they form a lattice rather than merely a join-semilattice. Spoken in terms of complexity classes, this means that the intersection of any two complexity classes admitting complete search problems will again have a complete search problem.

Second, there are some interesting applications where decision problems turn out to be completely inadequate for modeling. A typical example is algorithmic game theory studying tasks such as finding a Nash equilibrium of a game – as these are guaranteed to exist, but are not unique in general, the usual approaches to replace a search problem by a related decision problem fail here. From a more abstract point of view, [3] shows that under natural assumptions there are NP-search problems not reducible to their decision counterpart.

The reader may have noticed the absence of any commitment to either complexity theory or computability theory so far. This is justified by the fact that a lot of the theory of many-one reductions between search problems can be developed in a very abstract way, obliterating the need to specify any particular field. Considering search problems allows us to drop the principled distinction between the objects of interest and the witnesses for reductions, and to treat the latter as a special case of the former. In particular, search problems can be subjected to composition, unlike decision problems, which paves the way for category theoretic language to be employed.

Another difference between search problems and decision problems that affects the study of the corresponding degree structure is that while any given algorithm can only decide a single decision problem, it will solve various search problems. For example, the trivial search problem where anything is a solution to any instance is clearly solved by any terminating algorithm. This means that different problems f_1 , f_2 may be reducible to some g such that the witnesses for the reductions are identical. This forces us to take into consideration an entailment relation between search problems telling us whether solving f already means that we have solved g , too.

1.1 Overview

Section 2 is the main part of the present paper, as it provides a generic definition of many-one reducibility in a category theoretic framework, and contains some general structural results about the induced degree structures. Here Subsection 2.1 provides the technical details of the definitions, Subsection 2.2 then contains Theorem 12, stating informally that the degree structure of any many-one reduction between search problems is a distributive lattice. The additional structure as a Kleene-algebra is covered in Subsection 2.3.

The reader not interested in the categorical framework might prefer to jump directly to Sections ??, ?? to find Theorem 12 applied in various settings. While we always find a distributive lattice with the same (more or less) operations as infimum and supremum, other aspects of the degree-structure do depend on the setting. A few examples are given, and some open questions are posed regarding these aspects.

1.2 Preceding work

The complexity theory of search problems is dominated by practical examples forming complexity classes such as PLS (Polynomial Local Search) [22] and PPAD (Polynomial Parity

Argument, Directed) [31]. The complexity class PPAD has several problems from algorithmic game theory as complete problems, such as finding a Nash equilibrium in a bimatrix game [10, 11, 7, 8] (see also [32] for an overview and [16] for more general results). PLS also appears often in algorithmic game theory (see e.g. [17]), but more generally covers the search for locally optimal solutions (for an example, see [15], for an overview of results [1]).

Very recently it was noticed [12] that the intersection $\text{PLS} \cap \text{PPAD}$ contains a plethora of interesting problems for which no better upper bound was known, hence drawing interest to this complexity class. In particular, [12, Theorem 2.2] states that $\text{PLS} \cap \text{PPAD}$ has a complete problem, which also follows as a corollary of our Theorem 12.

As already mentioned, the consideration of functions (as being distinct from decision problems) is usually avoided in computability theory, which is more or less unproblematic because functions are identified with their graph. An exception to this is the study of enumeration complexity or bounded queries [19], however, multi-valuedness is not included here¹.

The line of work responsible for developing the structural theory of many-one reductions generalized here, started in [36], [37] with the study of continuous many-one reductions between search problems on Cantor and Baire space. In [5], computable many-one reductions between such problems were considered, and [6], [34] identified the degree structure as a distributive lattice. The ideas underlying Subsection 2.3 were developed for this setting in [33], [21].

While the blend of category and recursion theory has a significant history (e.g. [27], [13], see the survey [9]), this does not include the study of reductions. On the other hand, categorical models of linear logic as studied in [29] admit certain similarities to the operations appearing in the present paper, however, have no connections to recursion or even complexity theory.

1.3 A first example

To demonstrate the basic ideas, we shall briefly discuss polynomial-time computable many-one reductions between (Type-1) search problems. In this subsection, a search problem shall be formalized by a relation $P \subseteq \{0, 1\}^* \times \{0, 1\}^*$. $(x, y) \in P$ signifies that x denotes an instance of the search problem P , and y denotes a solution to this instance. As we consider search problems to generalize partial function, we write $\text{dom}(P) := \{x \mid \exists y. (x, y) \in P\}$.

Solving a search problem P entails solving another problem P' (denoted by $P' \preceq P$), if every instance with a solution in P' also has a solution in P , and every solution in P is also a solution of P' :

$$P' \preceq P \Leftrightarrow \text{dom}(P') \subseteq \text{dom}(P) \wedge P \subseteq P'$$

An algorithm f solves a search problem P , if for any $x \in \text{dom}(P)$ it computes some $f(x)$ with $P(x, f(x))$, i.e. if $P \preceq \text{Graph}(f)$ holds. This clearly demonstrates that any given algorithm solves various search problems, namely all those that are \preceq -below it.

A reduction from P to Q takes an instance of P , computes an instance of Q from it, and obtains a solution to Q for it. Then the original instance for P , together with the solution from Q have to be sufficient to compute the solution to P . Hence, the definition of many-one reductions will involve the usual pairing function $\langle \rangle : \{0, 1\}^* \times \{0, 1\}^* \rightarrow \{0, 1\}^*$.

$$P \leq_m Q \Leftrightarrow \exists h, k \text{ poly-time computable,} \\ \text{dom}(P) \subseteq \text{dom}(k) \wedge ((k(x), y) \in Q \Rightarrow (x, h(\langle x, y \rangle)) \in P)$$

¹Even the article [25] does not deal with multi-valued functions, despite the title being *Effective Search Problems*.

If we identify the witnesses h, k with their graphs² H, K , we can restate the definition in terms of the composition of search problems, defined via $P \circ Q = \{(x, z) \mid \exists y (x, y) \in Q \wedge (y, z) \in P\}$, together with the entailment relation \preceq and the diagonal function defined via $\Delta(x) = \langle x, x \rangle$:

$$P \leq_m Q \Leftrightarrow \exists H, K \text{ poly-time computable } P \preceq H \circ \langle \text{id}, Q \circ K \rangle \circ \Delta$$

Now one may go on to prove \leq_m to be a preorder, which essentially depends on polynomial-time computable functions to be closed under composition and product, and the entailment relation \preceq to respect these operations. In other words, the polynomial-time computable functions form a subcategory of the category of search problems.

In the next step, the operations \oplus and \sqcup on search problems are introduced via $P \oplus Q(\langle x, y \rangle) = 0P(x) \cup 1Q(y)$, $P \sqcup Q(0x) = P(x)$ and $P \sqcup Q(1x) = Q(x)$. Informally, $P \oplus Q$ allows us to ask a question to P and one to Q , and presents us with an answer for either of them; while $P \sqcup Q$ allows us to determine whether P or Q is invoked. These operations take on the rôle of infimum (\oplus) and supremum (\sqcup) regarding \leq_m , turning the corresponding degree structure into a (distributive) lattice.

2 A generic definition of many-one reductions

2.1 The category-theoretic foundations

The framework in which we will introduce many-one reductions is given by poset-enriched p-categories [35] with a few additional properties. P-categories were introduced to capture the rôle of the cartesian product (i.e. the pairing function $\langle \rangle$) defined on partial functions, where it no longer coincides with the categorical product. The partial order defined on the sets of morphisms between fixed objects corresponds to the entailment relation \preceq .

Definition 1 ([35]). A p-category is a category \mathcal{C} together with a naturally associative and naturally commutative bifunctor $\times : \mathcal{C} \times \mathcal{C} \rightarrow \mathcal{C}$ (the product³), a natural transformation Δ (the diagonal) between the identity functor and the derived functor $X \mapsto X \times X$, and two families of natural transformations $(\pi_1^A)_{A \in \text{Ob}(\mathcal{C})}$ and $(\pi_2^B)_{B \in \text{Ob}(\mathcal{C})}$ (the projections) where π_1^A is between the derived functor $X \mapsto X \times A$ and the identity, while π_2^B is between the derived functor $X \mapsto B \times X$ and the identity, such that the following properties are given:

$$\begin{aligned} \pi_1^X(X) \circ \Delta(X) &= \pi_2^X(X) \circ \Delta(X) = \text{id}_X & (\pi_1^Y(X) \times \pi_2^X(Y)) \circ \Delta(X \times Y) &= \text{id}_{X \times Y} \\ \pi_1^Y(X) \circ (\text{id}_X \times \pi_1^Z(Y)) &= \pi_1^{(Y \times Z)}(X) & \pi_1^Z(X) \circ (\text{id}_X \times \pi_2^Y(Z)) &= \pi_1^{(Y \times Z)}(X) \\ \pi_2^X(Y) \circ (\pi_1^Y(X) \times \text{id}_Z) &= \pi_2^{(X \times Y)}(Z) & \pi_2^X(Z) \circ (\pi_2^Y(Y) \times \text{id}_Z) &= \pi_2^{(X \times Y)}(Z) \end{aligned}$$

For easier reading, we shall write $\pi_1^{X,Y}$ instead of $\pi_1^Y(X)$, $\pi_2^{X,Y}$ for $\pi_2^X(Y)$ and finally Δ_X in place of $\Delta(X)$. If the superscripts are obvious from the context, they may be dropped.

The treatment of partial maps in a categorical framework causes the concept of the domain of a map to split into two separate ones. With $\text{Dom}(f)$ we denote the object A , if $f : A \rightarrow B$ is

²The identification of a function with its graph as a search problem is different from the identification of a function with its graph as a set, when it comes to algorithmic considerations.

³We hope that this nomenclature will not cause confusion, and point out again that the product in a p-category is not necessarily the product in the underlying category. It will be made clear whenever we refer to the categorical product instead.

a morphism. Following [13], we write $\text{dom}(f)$ for the morphism $\pi_1^{A,B} \circ (\text{id}_A \times f) \circ \Delta_A$, where π_1 is the first projection of the product $X \times Y$. One can interpret $\text{dom}(f) : A \rightarrow A$ as the partial identity on that part of A where the partial map f is actually defined.

On morphisms of the form $\text{dom } f : X \rightarrow X$, a partial order \subseteq may be defined via $(\text{dom } f) \subseteq (\text{dom } g)$, if $(\text{dom } f) \circ (\text{dom } g) = \text{dom } f$. This partial order even is a meet-semilattice, with composition \circ taking the rôle of the infimum representing the intersection. For $f : A \rightarrow B$ and $g : A \rightarrow C$, the morphism $g \circ \text{dom}(f)$ is the restriction of g to the domain of f , so we could introduce the extension ordering for partial maps via $f \subseteq g$, if $f = g \circ \text{dom}(f)$. Hence, any p-category may be considered to be poset enriched in a canonic way. However, we do not wish to restrict ourselves to the case where \preceq coincides with \subseteq (compare the example in Subsection 1.3).

The concept of domains and restrictions allows us to formulate how projections work together with diagonals, as we find for any morphisms $f : X \rightarrow Y$, $g : X \rightarrow Z$:

$$\pi_2^{Y,Z} \circ (f \times g) \circ \Delta_X = g \circ \text{dom}(f) \quad \text{and} \quad \pi_1^{Y,Z} \circ (f \times g) \circ \Delta_X = f \circ \text{dom}(g)$$

In a slightly more general situation, we can omit the diagonal and find:

$$\pi_1^{\text{CDom}(f), \text{CDom}(g)} \circ (f \times g) = f \circ \pi_1^{\text{Dom}(f), \text{Dom}(g)} \circ (\text{id}_{\text{Dom}(f)} \times \text{dom}(g))$$

Instead of taking \preceq to be the poset-enrichment, we define a poset enriched p-category to be a p-category (with underlying category \mathcal{C} and product \times) together with a family of partial orders $(\preceq_{A,B})_{A,B \in \text{Ob}(\mathcal{C})}$, where $\preceq_{A,B}$ is a partial order on the hom-set⁴ $\mathcal{C}(A,B)$, such that \preceq is compatible with both composition and product, i.e. $f \preceq g$ implies $f \circ h \preceq g \circ h$, $h \circ f \preceq h \circ g$ and $f \times h \preceq g \times h$ for suitable morphisms h . We also require that \preceq is in a specific sense well-behaved on domains, as we require that for any morphisms f, g, h such that $\text{dom}(f) \subseteq \text{dom}(g) \subseteq \text{dom}(h)$ if $\text{dom}(h) \preceq \text{dom}(f)$ holds, then $\text{dom}(h) \preceq \text{dom}(g)$ follows. Due to the compatibility of \preceq and \circ , we find $\text{dom}(g) \preceq \text{dom}(f)$ in this case without additional assumptions, hence, we see a certain splitting behaviour of domains under \preceq .

In a meet-semilattice enriched p-category \mathcal{C} we require all binary infima in $(\mathcal{C}(A,B), \preceq)$ to exist, and moreover, the infima to be compatible with composition and products.

The requirement for \preceq to be compatible with compositions and products also implies an interesting connection between \preceq and domains, again without additional assumptions:

Proposition 2. For any morphisms $f, g : X \rightarrow Y$ in a poset enriched p-category, $f \preceq g$ implies $\text{dom}(f) \preceq \text{dom}(g)$.

Proof. From $f \preceq g$ we can conclude $(\text{id}_X \times f) \preceq (\text{id}_X \times g)$, which in turn implies:

$$\left(\pi_1^{X,Y} \circ (\text{id} \times f) \circ \Delta_X \right) \preceq \left(\pi_1^{X,Y} \circ (\text{id} \times g) \circ \Delta_X \right)$$

Evaluation of both sides of this statement yields the claim. \square

While the presence of the product is sufficient to define many-reductions, coproducts are required for our results regarding the induced degree structure. We say that a poset-enriched p-category has α -coproducts for some cardinal α , if the underlying category has α -coproducts,

⁴If the category \mathcal{C} is not locally small, i.e. if $\mathcal{C}(A,B)$ might be a proper class, one might replace posets with partially ordered classes here.

and if these are compatible with both \times and \preceq , i.e. if there is a natural isomorphism $a : A \times (\coprod_{\nu < \alpha} B_\nu) \rightarrow \coprod_{\nu < \alpha} (A \times B_\nu)$ and $f_\nu \preceq g_\nu$ for all $n < \alpha$ implies $(\coprod_{\nu < \alpha} f_\nu) \preceq (\coprod_{\nu < \alpha} g_\nu)$.

The coproduct injections are denoted by $\iota_\mu^{(A_\nu)_{\nu < \alpha}} : A_\mu \rightarrow (\coprod_{\nu < \alpha} A_\nu)$. The co-diagonal is written as $\nabla_A^\alpha : (\coprod_{\nu < \alpha} A) \rightarrow A$, it satisfies $\nabla_A^\alpha \circ \iota_\mu^{(A_\nu)_{\nu < \alpha}} = \text{id}_A$ for all $\mu < \alpha$. More generally, we want all coproduct injections to be retractable. For this, assume that $Ob(\mathcal{C})$ is totally connected, i.e. for any two objects $A, B \in Ob(\mathcal{C})$ there is some morphism $c_{A,B} : A \rightarrow B$ in \mathcal{C} . Then a retract $\kappa_\mu^{(A_\nu)_{\nu < \alpha}}$ for the injection $\iota_\mu^{(A_\nu)_{\nu < \alpha}}$ can be obtained as $\kappa_\mu^{(A_\nu)_{\nu < \alpha}} = \nabla_{A_\mu}^\alpha \circ (\coprod_{\nu < \alpha} c_{A_\nu, A_\mu})$, when we assume $c_{A,A} = \text{id}_A$.

We shall need some facts about coproducts in p-categories:

Proposition 3. For any $X \in Ob(\mathcal{C})$ in a p-category with α -coproducts we find $\text{dom}(\nabla_X^\alpha) = \text{id}_X$.

Proof. We have to show $\pi_2^{X, (\coprod_{\nu < \alpha} X)} \circ (\nabla_X^\alpha \times \text{id}_{(\coprod_{\nu < \alpha} X)}) \circ \Delta_{(\coprod_{\nu < \alpha} X)} = \text{id}_X$. Due to the uniqueness condition for coproducts, this is equivalent to $\left[\pi_2^{X, (\coprod_{\nu < \alpha} X)} \circ (\nabla_X^\alpha \times \text{id}_{(\coprod_{\nu < \alpha} X)}) \circ \Delta_{(\coprod_{\nu < \alpha} X)} \right] \circ \iota_\mu^{(X)_{\nu < \alpha}} = \iota_\mu^{(X)_{\nu < \alpha}}$ for all $\mu < \alpha$. By naturality of the diagonal and \times being a functor, we get: $\pi_2^{X, (\coprod_{\nu < \alpha} X)} \circ (\text{id}_X \times \iota_\mu^{(X)_{\nu < \alpha}}) \circ \Delta_X = \iota_\mu^{(X)_{\nu < \alpha}}$, which is true. \square

Proposition 4. For any family $(X_\nu)_{\nu < \alpha}$, $X_\nu \in Ob(\mathcal{C})$ in a p-category with α -coproducts we find $\text{dom}(\iota_\mu^{(X_\nu)_{\nu < \alpha}}) = \text{id}_{X_\mu}$ for any $\mu < \alpha$.

Proof. The definition of coproducts requires $(\coprod_{\nu < \alpha} \text{id}_{X_\nu}) \circ \iota_\mu^{(X_\nu)_{\nu < \alpha}} = \text{id}_{X_\mu}$. Composition with $\text{dom}(\iota_\mu^{(X_\nu)_{\nu < \alpha}})$ from the right yields $(\coprod_{\nu < \alpha} \text{id}_{X_\nu}) \circ \iota_\mu^{(X_\nu)_{\nu < \alpha}} = \text{dom}(\iota_\mu^{(X_\nu)_{\nu < \alpha}})$, comparison of the two right sides provides the claim. \square

Proposition 5. For families $(f_\nu : X_\nu \rightarrow Y_\nu)_{\nu < \alpha}$, $(g_\nu : X_\nu \rightarrow Z)_{\nu < \alpha}$ of morphisms in a p-category with α -coproducts the following identity holds:

$$\nabla_{(\coprod_{\nu < \alpha} (Y_\nu \times Z))}^\alpha \circ a \circ \left[\left(\coprod_{\nu < \alpha} f_\nu \right) \times \left(\coprod_{\mu < \alpha} g_\mu \right) \right] \circ \Delta_{(\coprod_{\nu < \alpha} X_\nu)} = \coprod_{\nu < \alpha} ((f_\nu \times g_\nu) \circ \Delta_{X_\nu})$$

where $a : [(\coprod_{\nu < \alpha} Y_\nu) \times (\coprod_{\mu < \alpha} Z)] \rightarrow [\coprod_{\mu < \alpha} (\coprod_{\nu < \alpha} (Y_\nu \times Z))]$ is the canonic distributivity isomorphism.

Proof. It is sufficient to show that both sides of the equation are identical when composed with an arbitrary coproduct injection $\iota_\eta^{(X_\nu)_{\nu < \alpha}}$ for $\eta < \alpha$ from the right, hence we want to prove:

$$\nabla_{(\coprod_{\nu < \alpha} (Y_\nu \times Z))}^\alpha \circ a \circ \left[\left(\coprod_{\nu < \alpha} f_\nu \right) \times \left(\coprod_{\mu < \alpha} g_\mu \right) \right] \circ \Delta_{(\coprod_{\nu < \alpha} X_\nu)} \circ \iota_\eta^{(X_\nu)_{\nu < \alpha}} = \iota_\eta^{(Y_\nu \times Z)_{\nu < \alpha}} \circ ((f_\eta \times g_\eta) \circ \Delta_{X_\eta})$$

As Δ is a natural transformation and \times a functor, the injection $\iota_\eta^{(X_\nu)_{\nu < \alpha}}$ can be moved inwards to yield:

$$\nabla_{(\coprod_{\nu < \alpha} (Y_\nu \times Z))}^\alpha \circ a \circ \left[(\iota_\eta^{(Y_\nu)_{\nu < \alpha}} \circ f_\eta) \times (\iota_\eta^{(Z)_{\nu < \alpha}} \circ g_\eta) \right] \circ \Delta_{X_\eta} = \iota_\eta^{(Y_\nu \times Z)_{\nu < \alpha}} \circ ((f_\eta \times g_\eta) \circ \Delta_{X_\eta})$$

In the next step, the isomorphism a is taken into account:

$$\nabla_{(\coprod_{\nu < \alpha} (Y_\nu \times Z))}^\alpha \circ \iota_\eta^{(\coprod_{\nu < \alpha} (Y_\nu \times Z))_{\mu < \alpha}} \circ \iota_\eta^{(Y_\nu \times Z)_{\nu < \alpha}} \circ [f_\eta \times g_\eta] \circ \Delta_{X_\eta} = \iota_\eta^{(Y_\nu \times Z)_{\nu < \alpha}} \circ ((f_\eta \times g_\eta) \circ \Delta_{X_\eta})$$

The resulting identity follows directly from the definition of the co-diagonal. \square

Finally, a sub- poset enriched p-category of some poset enriched p-category with coproducts is a subcategory of the underlying category which is closed under products and coproducts, together with the suitable restrictions of the additional structures. It is wide, if it includes all objects and all domains (i.e. all morphisms of the form $\text{dom}(f)$) of the containing category. Clearly any sub- poset enriched p-category is a poset enriched p-category in its own right.

Now we have amassed the necessary building blocks to define the setting for our work:

Definition 6. A $(\alpha-)$ many-one category extension (moce) shall be a meet-semilattice-enriched p-category \mathcal{P} with α -coproducts, together with a wide and totally connected sub-poset enriched p-category \mathcal{S} .

In order to describe a moce we need four objects, hence we shall write them as quadruples $(\mathcal{P}, \mathcal{S}, \times, \preceq)$, where \mathcal{P} is a category made to a p-category by the functor \times , \preceq is a family of partial orders on the hom-sets of \mathcal{P} and \mathcal{S} is a suitable subcategory of \mathcal{P} made to a poset enriched p-category by the restrictions of \times and \preceq .

All standard examples for many-one reductions come from a poset-enriched p-category \mathcal{P} that is obtained from a category equivalent to a full subcategory of the category Rel of relations, with the cartesian product as product and $P' \preceq P \Leftrightarrow (\text{dom}(P') \subseteq \text{dom}(P) \wedge P \subseteq P')$ as partial order. However – even with some full subcategory of Rel as starting point – one could use the categorical product on Rel rather than the cartesian one, or the extension relationships \subseteq or \supseteq rather than the entailment order \preceq to obtain a poset enriched p-category⁵.

Occasionally we will be interested in special objects – or morphisms that behave sufficiently like objects, i.e. domains – in our categories. An initial object in a p-category is just an initial object in the underlying category, i.e. an object $I \in \text{Ob}(\mathcal{C})$ such that $|\mathcal{C}(I, A)| = 1$ for all $A \in \text{Ob}(\mathcal{C})$. We can generalize this notion to domains, by calling a domain $\text{dom } i$ initial, if for any $A \in \text{Ob}(\mathcal{C})$ there is exactly one morphism g with $g \circ \text{dom } i = g$ and $\text{CDom}(g) = A$. Clear I is initial, iff $\text{id}_I = \text{dom } \text{id}_I$ is initial as a domain.

Usually an object E in a category \mathcal{C} is called *empty*, if the existence of some morphism $g : A \rightarrow E$ implies that A is an initial object. Clearly, this definition is somewhat contradicted with our requirement for the relevant categories to be totally connected – only categories equivalent to the trivial category containing a single object and no further morphisms fulfills the criteria.

Calling a morphism g in a p-category total, if $\text{dom}(g) = \text{id}_{\text{Dom}(g)}$, we see that the total morphisms in a p-category form a sub-p-category, on which the p-category product even coincides with the categorical product. Now we define an empty object of a p-category to be an initial object in the underlying category which is empty in the subcategory of total morphisms. The concept of emptiness is extended to domains by calling an initial domain $\text{dom } e$ empty, if for any total morphism g with $\text{dom}(e) \circ g = g$ we find $\text{Dom}(g)$ to be an initial object.

Likewise, a final object of a p-category shall be a final object of the subcategory of total maps. A domain $\text{dom } f$ is called final, if for any object A there is exactly one total morphism g

⁵The compatibility requirement for the partial orders is self-dual, so reversing a suitable partial order would return a poset enriched p-category. Only the existence and compatibility of infima required for a moce might fail.

with $\text{dom}(f) \circ g = g$ and $\text{Dom}(g) = A$. Now we may conclude that $E \times A$ is isomorphic to E for any empty object E and any object A in a p-category, while for a final object F we find $F \times A$ to be isomorphic to A . These properties can be extended to domains.

Finally, we say that a domain is initial (empty, final) in a moce $(\mathcal{P}, \mathcal{S}, \times, \preceq)$, if it has this property in both \mathcal{P} and \mathcal{S} .

2.2 The lattice of many-one degrees

There are two definitions of many-one reductions commonly found in the literature on search or function problems, which differ in the question whether the post-processing of the oracle answer still has access to the input. Forgetting the input leads to a simpler definition, and may make proofs of non-reducibility easier, while retaining it yields the nicer degree structure and allows to formulate stronger and more meaningful separation statements. We shall speak of strong many-one reductions if the original input is forgotten, and of many-one reductions otherwise.

Throughout this subsection, we assume that some moce $(\mathcal{P}, \mathcal{S}, \times, \preceq)$ is given, and refrain from mentioning it explicitly where this is unnecessary.

Definition 7 (Strong many-one reductions). Let $f \leq_{sm} g$ hold for $f, g \in \mathcal{P}$, if there are $H, K \in \mathcal{S}$ with $f \preceq H \circ g \circ K$.

Proposition 8. (\mathcal{P}, \leq_{sm}) is a preordered class.

Proof. For any $f \in \mathcal{P}$, we have $\text{id}_{\text{Dom}(f)}, \text{id}_{\text{CDom}(f)} \in \mathcal{S}$. Trivially, $f = \text{id}_{\text{CDom}(f)} \circ f \circ \text{id}_{\text{Dom}(f)}$ holds. As \preceq is a preorder, this implies $f \leq_{sm} f$.

Now assume $f \leq_{sm} g$ and $g \leq_{sm} h$ witnessed by $H, K, F, G \in \mathcal{S}$. Due to the assumptions on \preceq , $g \preceq F \circ h \circ K$ implies $H \circ g \circ K \preceq (H \circ F) \circ h \circ (G \circ K)$. Transitivity of \preceq yields $f \preceq (H \circ F) \circ h \circ (G \circ K)$, hence $f \leq_{sm} h$ follows. \square

Definition 9 (Many-one reductions). Let $f \leq_m g$ hold for $f, g \in \mathcal{P}$, if there are $H, K \in \mathcal{S}$ with $f \preceq H \circ (\text{id}_{\text{Dom}(f)} \times (g \circ K)) \circ \Delta_{\text{Dom}(f)}$.

Proposition 10. $f \leq_{sm} g$ implies $f \leq_m g$.

Proof. As we require $\text{Ob}(\mathcal{P}) = \text{Ob}(\mathcal{S})$, and \mathcal{S} is closed under products, in particular we also have $\pi_2^{\text{Dom}(f), \text{CDom}(g)} \in \mathcal{S}$ for the respective projection. Now assume $f \preceq H \circ g \circ K$. Then also $f \preceq (H \circ \pi_2^{\text{Dom}(f), \text{CDom}(g)}) \circ (\text{id}_{\text{Dom}(f)} \times (g \circ K)) \circ \Delta_{\text{Dom}(f)}$ is true. \square

Proposition 11. (\mathcal{P}, \leq_m) is a preordered class.

Proof. Reflexivity of \leq_m follows from Propositions 8, 10. Now assume $f \leq_m g$ witnessed by $F, G \in \mathcal{S}$, and $g \leq_m h$ witnessed by $H, K \in \mathcal{S}$. Due to the assumptions on \preceq , $g \preceq H \circ (\text{id}_{\text{Dom}(g)} \times (h \circ K)) \circ \Delta_{\text{Dom}(g)}$ implies:

$$F \circ (\text{id}_{\text{Dom}(f)} \times (g \circ G)) \circ \Delta_{\text{Dom}(f)} \preceq F \circ (\text{id}_{\text{Dom}(f)} \times (H \circ (\text{id}_{\text{Dom}(g)} \times (h \circ K)) \circ \Delta_{\text{Dom}(g)} \circ G)) \circ \Delta_{\text{Dom}(f)}$$

Using the transitivity of \preceq and the naturality of the diagonal, one obtains:

$$f \preceq F \circ (\text{id}_{\text{Dom}(f)} \times (H \circ (G \times (h \circ K \circ G)) \circ \Delta_{\text{Dom}(f)})) \circ \Delta_{\text{Dom}(f)}$$

Now we use the distributivity of products over composition (i.e. the fact that \times is a functor):

$$f \preceq F \circ (\text{id}_{\text{Dom}(f)} \times H) \circ (\text{id}_{\text{Dom}(f)} \times ((G \times (h \circ K \circ G)) \circ \Delta_{\text{Dom}(f)})) \circ \Delta_{\text{Dom}(f)}$$

Then the associativity of products is used, with $a \in \mathcal{S}$ denoting the canonic isomorphism $a : ((A \times B) \times C) \rightarrow (A \times (B \times C))$ of suitable type:

$$f \preceq F \circ (\text{id}_{\text{Dom}(f)} \times H) \circ a \circ (((\text{id}_{\text{Dom}(f)} \times G) \circ \Delta_{\text{Dom}(f)}) \times (h \circ K \circ G)) \circ \Delta_{\text{Dom}(f)}$$

In the next step, again the distributivity of products over composition is relevant:

$$f \preceq F \circ (\text{id}_{\text{Dom}(f)} \times H) \circ a \circ (((\text{id}_{\text{Dom}(f)} \times G) \circ \Delta_{\text{Dom}(f)}) \times \text{id}_{\text{CDom}(h)}) \circ (\text{id}_{\text{Dom}(f)} \times (h \circ K \circ G)) \circ \Delta_{\text{Dom}(f)}$$

Now we abbreviate $M = F \circ (\text{id}_{\text{Dom}(f)} \times H) \circ a \circ (((\text{id}_{\text{Dom}(f)} \times G) \circ \Delta_{\text{Dom}(f)}) \times \text{id}_{\text{CDom}(h)})$ and $N = K \circ G$ and observe $M, N \in \mathcal{S}$, hence the following proves the remaining part of the claim:

$$f \preceq M \circ (\text{id}_{\text{Dom}(f)} \times (h \circ N)) \circ \Delta_{\text{Dom}(f)}$$

□

As usual, our interest is focused on the partial orders induced by the preorders (in particular by \leq_m) on their equivalence classes (or degrees). For any moce $(\mathcal{P}, \mathcal{S}, \times, \preceq)$, the partially ordered class of equivalence classes for \leq_m shall be denoted by $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$. The main result in this subsection is the following:

Theorem 12. $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$ is a distributive lattice.

The proof of Theorem 12 will be spread over the following lemmata, which also give category-theoretic descriptions of the suprema and infima in $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$.

Lemma 13. $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$ has α -suprema, and these are given by α -coproducts, i.e.:

$$\sup_{\leq_m, \nu < \alpha} f_\nu = \coprod_{\nu < \alpha} f_\nu$$

Proof. 1. $f_\lambda \leq_{sm} \coprod_{\nu < \alpha} f_\nu$ for all $\lambda < \alpha$

By assumption, $\iota_\lambda^{(\text{Dom}(f_\nu))_{\nu < \alpha}}, \kappa_\lambda^{(\text{CDom}(f_\nu))_{\nu < \alpha}} \in \mathcal{S}$ for the respective coproduct injections and retracts of coproduct injections. The claim then follows from the following equation:

$$f_\lambda = \kappa_\lambda^{(\text{CDom}(f_\nu))_{\nu < \alpha}} \circ \left(\coprod_{\nu < \alpha} f_\nu \right) \circ \iota_\lambda^{(\text{Dom}(f_\nu))_{\nu < \alpha}}$$

2. $f_\lambda \leq_m \coprod_{\nu < \alpha} f_\nu$ for all $\lambda < \alpha$

Follows from 1. via Proposition 10.

3. $\coprod_{\nu < \alpha} g \leq_m g$

Let $a : (\coprod_{\nu < \alpha} \text{Dom}(g)) \times \text{CDom}(g) \rightarrow \text{Dom}(g) \times (\coprod_{\nu < \alpha} \text{CDom}(g))$ be the canonic isomorphism due to the distributive nature of α -coproducts and products⁶. We use

$\nabla_{\text{Dom}(g)}^\alpha, a, \pi_2^{\text{Dom}(g), (\coprod_{\nu < \alpha} \text{CDom}(g))} \in \mathcal{S}$. Hence, the following equation proves the claim:

$$\coprod_{\nu < \alpha} g = (\pi_2^{\text{Dom}(g), (\coprod_{\nu < \alpha} \text{CDom}(g))} \circ a) \circ \left[\text{id}_{(\coprod_{\nu < \alpha} \text{Dom}(g))} \times (g \circ \nabla_{\text{Dom}(g)}^\alpha) \right] \circ \Delta_{(\coprod_{\nu < \alpha} \text{Dom}(g))}$$

⁶To be more precise, in order to construct this isomorphism we need to invoke the distributivity law for α -coproducts and products twice, as well as the commutativity law for products.

To derive this equation, first consider the effect of the isomorphism a :

$$\coprod_{\nu < \alpha} g = \pi_2^{\text{Dom}(g), (\coprod_{\nu < \alpha} \text{CDom}(g))} \circ \left[\nabla_{\text{Dom}(g)}^\alpha \times \left(\coprod_{\nu < \alpha} g \right) \right] \circ \Delta_{(\coprod_{\nu < \alpha} \text{Dom}(g))}$$

The interaction of projections and diagonals makes this equivalent to:

$$\coprod_{\nu < \alpha} g = \left(\coprod_{\nu < \alpha} g \right) \circ \text{dom}(\nabla_{\text{Dom}(g)}^\alpha)$$

The latter equation follows directly from Proposition 3.

4. $f_\nu \leq_m g$ for all ν implies $(\coprod_{\nu < \alpha} f_\nu) \leq_m (\coprod_{\nu < \alpha} g)$.

Let $f_\nu \leq_m g$ be witnessed by H_ν, K_ν . Further let

$$a : \left(\coprod_{\nu < \alpha} \text{Dom}(f_\nu) \right) \times \left(\coprod_{\mu < \alpha} \text{CDom}(g) \right) \rightarrow \coprod_{\mu < \alpha} \left[\coprod_{\nu < \alpha} (\text{Dom}(f_\nu) \times \text{CDom}(g)) \right]$$

be the canonic isomorphism obtained from the distributivity law. We use ∇^α to abbreviate $\nabla_{[\coprod_{\nu < \alpha} (\text{Dom}(f_\nu) \times \text{CDom}(g))]}^\alpha$. Then $((\coprod_{\nu < \alpha} H_\nu) \circ \nabla^\alpha \circ a)$ and $(\coprod_{\nu < \alpha} K_\nu)$ witness the claim. For this, consider:

$$\left(\coprod_{\nu < \alpha} H_\nu \right) \circ \nabla^\alpha \circ a \circ \left[\text{id}_{(\coprod_{\nu < \alpha} \text{Dom}(f_\nu))} \times \left(\left(\coprod_{\nu < \alpha} g \right) \circ \left(\coprod_{\nu < \alpha} K_\nu \right) \right) \right] \circ \Delta_{(\coprod_{\nu < \alpha} \text{Dom}(f_\nu))}$$

Composition always commutes with coproducts of the same type:

$$\left(\coprod_{\nu < \alpha} H_\nu \right) \circ \nabla^\alpha \circ a \circ \left[\left(\coprod_{\nu < \alpha} \text{id}_{\text{Dom}(f_\nu)} \right) \times \left(\coprod_{\nu < \alpha} (g \circ K_\nu) \right) \right] \circ \Delta_{(\coprod_{\nu < \alpha} \text{Dom}(f_\nu))}$$

Now we can invoke Proposition 5 to obtain:

$$\left(\coprod_{\nu < \alpha} H_\nu \right) \circ \left[\coprod_{\nu < \alpha} ((\text{id}_{\text{Dom}(f_\nu)} \times (g \circ K_\nu)) \circ \Delta_{\text{Dom}(f_\nu)}) \right]$$

Invoking commutativity of coproducts and composition again, we get:

$$\coprod_{\nu < \alpha} [H_\nu \circ (\text{id}_{\text{Dom}(f_\nu)} \times (g \circ K_\nu)) \circ \Delta_{\text{Dom}(f)}]$$

As \preceq and α -coproducts commute, we know that $(\coprod_{\nu < \alpha} f_\nu)$ is \preceq -below the expression above. This concludes this part of the proof.

5. $f_\nu \leq_m g$ for all ν implies $(\coprod_{\nu < \alpha} f_\nu) \leq_m g$.

This follows by applying transitivity of \leq_m from Proposition 11 to 3. and 4.

6. The claim is equivalent to 2. and 5.

□

The infima are not given by a purely category-theoretic construction, but rather rely on the poset enrichment together with the assumption that in \mathcal{S} , suitable binary infima actually exist. As projections and injections are all included in \mathcal{S} by assumption, we can define the following:

Definition 14. For any morphisms $f, g \in \mathcal{P}$ define $(f \oplus g) : (\text{Dom}(f) \times \text{Dom}(g)) \rightarrow (\text{CDom}(f) \coprod \text{CDom}(g))$ via:

$$(f \oplus g) = \inf_{\preceq, i \in \{1,2\}} \{ \iota_i^{\text{CDom}(f), \text{CDom}(g)} \circ \pi_i^{\text{CDom}(f), \text{CDom}(g)} \} \circ (f \times g)$$

Lemma 15. $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$ has (binary) infima, and these are given by \oplus , i.e.:

$$\inf_{\leq_m} \{f, g\} = f \oplus g$$

Proof. 1. $(f \oplus g) \leq_{sm} f$ and $(f \oplus g) \leq_{sm} g$

As \oplus inherits commutativity from products and coproducts, it is sufficient to prove $(f \oplus g) \leq_{sm} f$. This is witnessed by $\iota_1^{\text{CDom}(f), \text{CDom}(g)}$ and $\left[\pi_1^{\text{Dom}(f), \text{Dom}(g)} \circ (\text{id}_{\text{Dom}(f)} \times \text{dom}(g)) \right]$, as we find:

$$\begin{aligned} & \iota_1^{\text{CDom}(f), \text{CDom}(g)} \circ f \circ \left[\pi_1^{\text{Dom}(f), \text{Dom}(g)} \circ (\text{id}_{\text{Dom}(f)} \times \text{dom}(g)) \right] \\ &= \iota_1^{\text{CDom}(f), \text{CDom}(g)} \circ \pi_1^{\text{CDom}(f), \text{CDom}(g)} \circ (f \times g) \end{aligned}$$

The latter expression is clearly \preceq -above $f \oplus g$, as can be verified from Definition 14.

2. $(f \oplus g) \leq_m f$, $(f \oplus g) \leq_m g$

Follows from 1. via Proposition 10.

3. If $h \leq_m f$ and $h \leq_m g$, then $h \leq_m (f \oplus g)$.

Let $h \leq_m f$ be witnessed by H_1, K_1 and let $h \leq_m g$ be witnessed by H_2, K_2 . Further let $a : [\text{Dom}(h) \times (\text{CDom}(f) \coprod \text{CDom}(g))] \rightarrow [(\text{Dom}(h) \times \text{CDom}(f)) \coprod (\text{Dom}(h) \times \text{CDom}(g))]$ be the canonic distributivity isomorphism. The claim now is witnessed by $\left[\nabla_{\text{CDom}(h)} \circ \left(\coprod_{i \in \{1,2\}} H_i \right) \circ a \right]$ and $[(K_1 \times K_2) \circ \Delta_{\text{Dom}(h)}]$. To prove this, we have to show that the following morphism is \preceq -above h :

$$\left[\nabla_{\text{CDom}(h)} \circ \left(\coprod_{i \in \{1,2\}} H_i \right) \circ a \right] \circ \left[\text{id}_{\text{Dom}(h)} \times \left(\inf_{\preceq, i \in \{1,2\}} \{ \iota_i^{\text{CDom}(f), \text{CDom}(g)} \circ \pi_i^{\text{CDom}(f), \text{CDom}(g)} \} \circ (f \times g) \circ [(K_1 \times K_2) \circ \Delta_{\text{Dom}(h)}] \right) \right] \circ \Delta_{\text{Dom}(h)}$$

As \inf and \circ are compatible, we can move the \inf -operator to the outside, and obtain a morphism that is \preceq -below the preceding one, hence it will suffice to show that h is \preceq -below the following:

$$\inf_{\preceq, i \in \{1,2\}} \left\{ \nabla_{\text{CDom}(h)} \circ \left(\coprod_{j \in \{1,2\}} H_j \right) \circ a \circ \left[\text{id}_{\text{Dom}(h)} \times \left(\iota_i^{\text{CDom}(f), \text{CDom}(g)} \circ \pi_i^{\text{CDom}(f), \text{CDom}(g)} \circ [(f \circ K_1) \times (g \circ K_2)] \circ \Delta_{\text{Dom}(h)} \right) \right] \circ \Delta_{\text{Dom}(h)} \right\}$$

Using the standard properties of the isomorphism a , coproducts, injections and the co-diagonal, this is equivalent to:

$$\inf_{\preceq, i \in \{1,2\}} \left\{ H_i \circ \left[\text{id}_{\text{Dom}(h)} \times \left(\pi_i^{\text{CDom}(f), \text{CDom}(g)} \circ [(f \circ K_1) \times (g \circ K_2)] \circ \Delta_{\text{Dom}(h)} \right) \right] \circ \Delta_{\text{Dom}(h)} \right\}$$

Applying the projections yields the following equivalent expression:

$$\inf_{\preceq} \left\{ [H_1 \circ (\text{id}_{\text{Dom}(h)} \times [f \circ K_1 \circ \text{dom}(g \circ K_2)]) \circ \Delta_{\text{Dom}(h)}], [H_2 \circ (\text{id}_{\text{Dom}(h)} \times [g \circ K_2 \circ \text{dom}(f \circ K_1)]) \circ \Delta_{\text{Dom}(h)}] \right\}$$

The domain-morphisms can be moved past the diagonal to arrive at:

$$\inf_{\preceq} \{ [H_1 \circ (\text{id}_{\text{Dom}(h)} \times [f \circ K_1]) \circ \Delta_{\text{Dom}(h)}] \circ \text{dom}(g \circ K_2), [H_2 \circ (\text{id}_{\text{Dom}(h)} \times [g \circ K_2]) \circ \Delta_{\text{Dom}(h)}] \circ \text{dom}(f \circ K_1) \}$$

By assumption, we have $h \preceq [H_2 \circ (\text{id}_{\text{Dom}(h)} \times [g \circ K_2]) \circ \Delta_{\text{Dom}(h)}]$, so from Proposition 2 we can conclude $\text{dom}(h) \preceq \text{dom}([H_2 \circ (\text{id}_{\text{Dom}(h)} \times [g \circ K_2]) \circ \Delta_{\text{Dom}(h)}])$. Straight-forward consideration shows $\text{dom}([H_2 \circ (\text{id}_{\text{Dom}(h)} \times [g \circ K_2]) \circ \Delta_{\text{Dom}(h)}]) \subseteq \text{dom}(g \circ K_2)$. By composition with $\text{dom}(h)$ from the right, we arrive at

$$\text{dom}([H_2 \circ (\text{id}_{\text{Dom}(h)} \times [g \circ K_2]) \circ \Delta_{\text{Dom}(h)}] \circ \text{dom}(h)) \subseteq \text{dom}(g \circ K_2 \circ \text{dom}(h)) \subseteq \text{dom}(h)$$

and $\text{dom}(h) \preceq \text{dom}([H_2 \circ (\text{id}_{\text{Dom}(h)} \times [g \circ K_2]) \circ \Delta_{\text{Dom}(h)}] \circ \text{dom}(h))$, so the splitting property of \preceq over \subseteq implies $\text{dom}(h) \preceq \text{dom}(g \circ K_2 \circ \text{dom}(h))$. If H_2, K_2 witnesses $h \preceq g$, then so do $H_2, (K_2 \circ \text{dom}(h))$, so w.l.o.g. we may assume $\text{dom}(g \circ K_2 \circ \text{dom}(h)) = \text{dom}(g \circ K_2)$, so we even find $\text{dom}(h) \preceq \text{dom}(g \circ K_2)$. But then $h \preceq [H_1 \circ (\text{id}_{\text{Dom}(h)} \times [f \circ K_1]) \circ \Delta_{\text{Dom}(h)}]$ implies:

$$h \preceq [H_1 \circ (\text{id}_{\text{Dom}(h)} \times [f \circ K_1]) \circ \Delta_{\text{Dom}(h)}] \circ \text{dom}(g \circ K_2)$$

By symmetry, we can use the same way to prove:

$$h \preceq [H_2 \circ (\text{id}_{\text{Dom}(h)} \times [g \circ K_2]) \circ \Delta_{\text{Dom}(h)}] \circ \text{dom}(f \circ K_1)$$

This concludes the proof of the claim.

4. 2. and 3. are the defining properties of the infimum. □

Lemma 16. $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$ is distributive, i.e. for all $f, g_\nu \in \mathcal{P}$:

$$f \oplus \left(\coprod_{\nu < \alpha} g_\nu \right) \equiv_m \coprod_{\nu < \alpha} (f \oplus g_\nu)$$

Proof. 1. $f \oplus (\coprod_{\nu < \alpha} g_\nu) \leq_{sm} \coprod_{\nu < \alpha} (f \oplus g_\nu)$

Let $a : [\text{Dom}(f) \times (\coprod_{\nu < \alpha} \text{Dom}(g_\nu))] \rightarrow [\coprod_{\nu < \alpha} (\text{Dom}(f) \times \text{Dom}(g_\nu))]$ be the canonic distributivity isomorphism. Further let $b : [\coprod_{\nu < \alpha} (\text{CDom}(f) \coprod \text{CDom}(g_\nu))] \rightarrow [\text{CDom}(f) \coprod (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))]$ be the canonic associativity isomorphism. Then a and b witness the reduction, i.e.:

$$\left[f \oplus \left(\coprod_{\nu < \alpha} g_\nu \right) \right] \preceq b \circ \left[\coprod_{\nu < \alpha} (f \oplus g_\nu) \right] \circ a$$

As a is an isomorphism, we can apply the inverse a^{-1} from the right on both sides and obtain an equivalent statement:

$$\left[f \oplus \left(\coprod_{\nu < \alpha} g_\nu \right) \right] \circ a^{-1} \preceq b \circ \left[\coprod_{\nu < \alpha} (f \oplus g_\nu) \right]$$

Now both sides have a coproduct as a domain, so by compatibility of \preceq and coproducts as well as composition, the statement above is equivalent to the one below holding for all $\mu < \alpha$:

$$\left[f \oplus \left(\coprod_{\nu < \alpha} g_\nu \right) \right] \circ a^{-1} \circ \iota_\mu^{(\text{Dom}(f) \times \text{Dom}(g_\nu))_{\nu < \alpha}} \preceq b \circ \left[\coprod_{\nu < \alpha} (f \oplus g_\nu) \right] \circ \iota_\mu^{(\text{Dom}(f) \times \text{Dom}(g_\nu))_{\nu < \alpha}}$$

This evaluates to:

$$\left[f \oplus \left(\coprod_{\nu < \alpha} g_\nu \right) \right] \circ (\text{id}_{\text{Dom}(f)} \times \iota_\mu^{(\text{Dom}(g_\nu))_{\nu < \alpha}}) \preceq b \circ \iota_\mu^{(\text{CDom}(f) \amalg \text{CDom}(g_\nu))_{\nu < \alpha}} \circ (f \oplus g_\mu)$$

To prove this statement, we insert the definition of \oplus and move the coproduct injection on the left side further to the left:

$$\inf_{\preceq, i \in \{1,2\}} \left\{ \iota_i^{\text{CDom}(f), (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))} \circ \pi_i^{\text{CDom}(f), (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))} \right\} \circ (\text{id}_{\text{CDom}(f)} \times \iota_\mu^{(\text{CDom}(g_\nu))_{\nu < \alpha}}) \circ [f \times g_\mu] \\ \preceq b \circ \iota_\mu^{(\text{CDom}(f) \amalg \text{CDom}(g_\nu))_{\nu < \alpha}} \circ \inf_{\preceq, i \in \{1,2\}} \left\{ \iota_i^{\text{CDom}(f), \text{CDom}(g_\mu)} \circ \pi_i^{\text{CDom}(f), \text{CDom}(g_\mu)} \right\} \circ (f \times g_\mu)$$

As we assume that composition preserves infima, we can move these to the outside. Additionally, we can drop the composition with $(f \times g_\mu)$ from both sides to arrive at a stronger statement:

$$\inf_{\preceq, i \in \{1,2\}} \left\{ \iota_i^{\text{CDom}(f), (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))} \circ \pi_i^{\text{CDom}(f), (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))} \circ (\text{id}_{\text{CDom}(f)} \times \iota_\mu^{(\text{CDom}(g_\nu))_{\nu < \alpha}}) \right\} \\ \preceq \inf_{\preceq, i \in \{1,2\}} \left\{ b \circ \iota_\mu^{(\text{CDom}(f) \amalg \text{CDom}(g_\nu))_{\nu < \alpha}} \circ \iota_i^{\text{CDom}(f), \text{CDom}(g_\mu)} \circ \pi_i^{\text{CDom}(f), \text{CDom}(g_\mu)} \right\}$$

On the left side, we can move the projections past the product due to $\text{dom}(\iota_\mu^{(\text{CDom}(g_\nu))_{\nu < \alpha}}) = \text{id}_{\text{CDom}(g_\mu)}$ (Proposition 4), and on the right side we take into consideration the effects of the isomorphism b :

$$\inf_{\preceq} \left\{ \left[\iota_1^{\text{CDom}(f), (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))} \circ \pi_1^{\text{CDom}(f), \text{CDom}(g_\mu)} \right], \left[\iota_2^{\text{CDom}(f), (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))} \circ \iota_\mu^{(\text{CDom}(g_\nu))_{\nu < \alpha}} \circ \pi_2^{\text{CDom}(f), \text{CDom}(g_\mu)} \right] \right\} \\ \preceq \inf_{\preceq} \left\{ \left[\iota_1^{\text{CDom}(f), (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))} \circ \pi_1^{\text{CDom}(f), \text{CDom}(g_\mu)} \right], \left[\iota_2^{\text{CDom}(f), (\coprod_{\nu < \alpha} \text{CDom}(g_\nu))} \circ \iota_\mu^{(\text{CDom}(g_\nu))_{\nu < \alpha}} \circ \pi_2^{\text{CDom}(f), \text{CDom}(g_\mu)} \right] \right\}$$

As both sides are identical, this statement true, and, as shown above, implies our claim.

$$2. f \oplus \left(\coprod_{\nu < \alpha} g_\nu \right) \leq_m \coprod_{\nu < \alpha} (f \oplus g_\nu)$$

Follows from 1. via Proposition 10.

$$3. \coprod_{\nu < \alpha} (f \oplus g_\nu) \leq_m f \oplus \left(\coprod_{\nu < \alpha} g_\nu \right)$$

This direction holds in every lattice, hence, it follows from Theorem 12.

□

Proposition 17. If $(\mathcal{P}, \mathcal{S}, \times, \preceq)$ has an initial domain, then $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$ has a bottom element.

Proof. Let $i = \text{dom } i : I \rightarrow I$ be an initial domain in \mathcal{S} . We claim $i \leq_{sm} f$ for any $f \in \mathcal{P}$. By Proposition 10 this implies the original statement. As \mathcal{S} is totally connected, there is some morphism $c_{\text{CDom}(f), I} : \text{CDom}(f) \rightarrow I$ in \mathcal{S} . Also, there is a morphism $c_{I, \text{Dom}(f)} : I \rightarrow \text{Dom}(f)$ satisfying $c_{I, \text{Dom}(f)} \circ i = c_{I, \text{Dom}(f)}$, as i is initial. Then $c_{\text{CDom}(f), I} \circ f \circ c_{I, \text{Dom}(f)} : I \rightarrow I$ must be equal to i , as we have both $i \circ i = i$ as well as $c_{\text{CDom}(f), I} \circ f \circ c_{I, \text{Dom}(f)} \circ i = c_{\text{CDom}(f), I} \circ f \circ c_{I, \text{Dom}(f)}$. □

2.3 The Kleene-algebra of many-one degrees

Besides the lattice-structure, the many-one degrees also have the structure of a Kleene-algebra⁷[24], provided certain conditions are fulfilled. We shall start by discussing the underlying idempotent semiring. The addition in the Kleene-algebra is the supremum of the lattice, i.e. the coproduct. The multiplication in the Kleene-algebra is induced by the product of the p-category, as to be shown next.

Lemma 18. \times induces an operation on $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$, i.e. $f_i \leq_m g_i$ for $i \in \{1, 2\}$ implies $(f_1 \times f_2) \leq_m (g_1 \times g_2)$.

Proof. Let $f_i \leq_m g_i$ be witnessed by H_i, K_i , and let $a : [(\text{Dom}(f_1) \times \text{Dom}(f_2)) \times (\text{CDom}(g_1) \times \text{CDom}(g_2))] \rightarrow [(\text{Dom}(f_1) \times \text{CDom}(g_1)) \times (\text{Dom}(f_2) \times \text{CDom}(g_2))]$ be the canonic isomorphism constructed from associativity and commutativity of \times . Then $(H_1 \times H_2) \circ a$ and $(K_1 \times K_2)$ witnesses $(f_1 \times f_2) \leq_m (g_1 \times g_2)$ as can be seen from

$$(H_1 \times H_2) \circ a \circ [\text{id}_{\text{Dom}(f_1) \times \text{Dom}(f_2)} \times ((g_1 \times g_2) \circ (K_1 \times K_2))] \circ \Delta_{\text{Dom}(f_1) \times \text{Dom}(f_2)}$$

being equal to

$$[H_1 \circ (\text{id}_{\text{Dom}(f_1)} \times (g_1 \circ K_1)) \circ \Delta_{\text{Dom}(f_1)}] \times [H_2 \circ (\text{id}_{\text{Dom}(f_2)} \times (g_2 \circ K_2)) \circ \Delta_{\text{Dom}(f_2)}]$$

together with the fact that \times is compatible with \preceq . \square

For completeness, we shall also state the following, while omitting the trivial proof:

Lemma 19. \times induces an operation on the \leq_{sm} -degrees, i.e. $f_i \leq_{sm} g_i$ for $i \in \{1, 2\}$ implies $(f_1 \times f_2) \leq_{sm} (g_1 \times g_2)$.

Theorem 20. Let the p-category \mathcal{P} have an empty domain e and a final domain f . Then the degrees $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$ together with the operations \coprod and \times and the constants $E, F \in \mathfrak{D}$ form an idempotent commutative semiring, i.e. the following hold for all $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathfrak{D}$:

1. $\mathbf{a} \coprod \mathbf{a} = \mathbf{a}$, $(\mathbf{a} \coprod \mathbf{b}) \coprod \mathbf{c} = \mathbf{a} \coprod (\mathbf{b} \coprod \mathbf{c})$, $\mathbf{a} \coprod \mathbf{b} = \mathbf{b} \coprod \mathbf{a}$
2. $\mathbf{a} \coprod e = \mathbf{a}$
3. $(\mathbf{a} \times \mathbf{b}) \times \mathbf{c} = \mathbf{a} \times (\mathbf{b} \times \mathbf{c})$, $\mathbf{a} \times \mathbf{b} = \mathbf{b} \times \mathbf{a}$
4. $\mathbf{a} \times f = \mathbf{a}$, $\mathbf{a} \times e = e$
5. $\mathbf{a} \times (\mathbf{b} \coprod \mathbf{c}) = (\mathbf{a} \times \mathbf{b}) \coprod (\mathbf{a} \times \mathbf{c})$

Proof. First of all, note that by Lemmata 13, 18 the operations are well-defined. Further, by Proposition 10 morphisms in \mathcal{P} that are isomorphic over \mathcal{S} represent the same degree in $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$. Hence, 1. follows from Lemma 13. As every empty domain is initial, 2. follows from Lemma 13 and Proposition 17. Claim 3. is a direct consequence of associativity and commutativity of the functor \times , while 4. comes from the interaction of \times with special objects or domains. Finally, 5. is implied by the requirement that the functor \times distributes over coproducts. \square

⁷We are referring to the algebraic concept here, not to the distributive lattice with an involution!

The product can be iterated: We inductively define $f^1 = f$, $f^{n+1} = f^n \times f$ for any morphism, and follow by $f^* = \coprod_{n \in \mathbb{N}} f^n$, assuming this coproduct exists. Intuitively, access to f^* allows us to use f for any predetermined finite number of times in parallel; hence, the operation $*$ is suitable to introduce a generalization of *wtt*-degrees in our framework.

In a \aleph_0 -moce, we have the prerequisites to deal with all countable coproducts, in particular with f^* . However, our computability-inspired examples do not fulfill the respective criteria: The category of computable functions is not closed under countable coproducts. However, forming countable coproducts is usually unproblematic in the super-category \mathcal{P} , and sufficiently uniform countable coproducts, such as f^* even preserve computability. We will prove the existence of the Kleene-algebra structure assuming that \mathcal{S} has all needed coproducts; so in order to apply the result, their presence has to be checked individually.

As preparation, we point out that for any two objects $X, Y \in \text{Ob}(\mathcal{S})$ we find $X^* \times Y^*$ to be isomorphic to $(X \times Y)^* \coprod \left(\coprod_{n, m \in \mathbb{N}, n \neq m} (X^n \times Y^m) \right)$, hence, we have a retractable embedding $(X \times Y)^* \hookrightarrow (X^* \times Y^*)$ in \mathcal{S} due to the assumption that \mathcal{S} is totally connected. Further, we need to consider objects of the form $(X^*)^m$. Iterating the distributivity of products over coproducts, we obtain an isomorphism $a_m : (X^*)^m \rightarrow \coprod_{n_1, \dots, n_m \in \mathbb{N}} X^{n_1 + \dots + n_m}$ in \mathcal{S} . For any $n \in \mathbb{N}$, we let $p(n)$ denote the number of distinct order-depending summations of the form $n = n_1 + \dots + n_m$ with varying $m \in \mathbb{N}$. Then we obtain an isomorphism $a : (X^*)^* \rightarrow \coprod_{n \in \mathbb{N}} \left(\coprod_{i \leq p(n)} X^n \right)$ in \mathcal{S} by taking first the coproduct over all a_m , and then rearranging the resulting coproduct. Motivated by this, we will use $\nabla_X^* : (X^*)^* \rightarrow X^*$ to denote the composition $\nabla_X^* = \left(\coprod_{n \in \mathbb{N}} \nabla_X^{p(n)} \right) \circ a$. Furthermore, consider the product $\left(\coprod_{n \in \mathbb{N}} \left(\coprod_{i \leq p(n)} X^n \right) \right) \times \left(\coprod_{m \in \mathbb{N}} Y^m \right)$. Due to distributivity and associativity, this is isomorphic to $\left[\coprod_{n \in \mathbb{N}} \coprod_{i \leq p(n)} (X^n \times Y^n) \right] \coprod \left[\coprod_{n, m \in \mathbb{N}, n \neq m} \coprod_{i \leq p(n)} (X^n \times Y^m) \right]$. Compose the respective isomorphisms with a retract to the first component of the final coproduct to obtain a canonic morphism $c : (X^*)^* \times Y^* \rightarrow \left[\coprod_{n \in \mathbb{N}} \coprod_{i \leq p(n)} (X^n \times Y^n) \right]$. Finally, apply $\left(\coprod_{n \in \mathbb{N}} \pi_1^{X^n, Y^n} \right)$ and another isomorphism to obtain the canonic morphism $\chi_{X, Y} : ((X^*)^* \times Y^*) \rightarrow (Y^*)^*$.

Theorem 21. Given an \aleph_0 -moce $(\mathcal{P}, \mathcal{S}, \times, \leq)$, $*$ induces a closure operator on $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \leq)$, i.e. for all $f, g \in \mathcal{P}$:

1. $f \leq_m f^*$
2. $f \leq_m g$ implies $f^* \leq_m g^*$
3. $(f^*)^* \leq_m f^*$

Proof. 1. $f \leq_m f^*$

This follows directly from Lemma 13 (2) together with the definition of $*$.

2. $f \leq_m g$ implies $f^* \leq_m g^*$

Let $f \leq_m g$ be witnessed by $H, K \in \mathcal{S}$. Let $a \in \mathcal{S}$ be an retract of the embedding $(\text{Dom}(f)^* \times \text{CDom}(g)^*) \hookrightarrow (\text{Dom}(f) \times \text{CDom}(g))^*$. Then $(H^* \circ a)$ and K^* witness the claim.

3. $(f^*)^* \leq_m f^*$

The witnesses are $\nabla_{\text{Dom}(f)}^*$ and $\chi_{\text{Dom}(f), \text{CDom}(f)}$.

□

Corollary 22. Let the p-category \mathcal{P} have an empty object E and a final object F . Then $(\mathfrak{D}, \coprod, \times, E, F, *)$ is a continuous Kleene-algebra.

An important consequence of Theorem 21 is that the following actually defines a preorder:

Definition 23. Let Theorem 21 hold for $\mathfrak{D}(\mathcal{P}, \mathcal{S}, \times, \preceq)$. For $f, g \in \mathcal{P}$, define $f \leq_{wtt} g$, if $f \leq_m g^*$, or equivalently $f^* \leq_m g^*$ holds. The resulting degrees are denoted by \mathfrak{D}^* .

Corollary 24. \mathfrak{D}^* is a lattice, and a sub-meet-semilattice of \mathfrak{D} .

3 Standard Examples

The standard examples of many-one reducibilities are all derived from a moce $(\mathcal{P}, \mathcal{S}, \times, \preceq)$ where \mathcal{P} is some full subcategory of the category Rel of relations (or multi-valued functions), which is made into a p-category via the cartesian product \times ; while the partial order \preceq is defined via $P' \preceq P \Leftrightarrow \text{dom}(P') \subseteq \text{dom}(P) \wedge P \subseteq P'$. Once we have verified that these fulfill our axioms, any specific reducibility is specified by a subcategory of Rel closed under products and coproducts, and Theorem 12 becomes immediately applicable.

We shall fill in the details now. The objects of Rel are just the sets, and the morphisms in $Rel(A, B)$ are all $f \subseteq A \times B$. Composition is defined for $f : A \rightarrow B$, $g : C \rightarrow A$ as $f \circ g = \{(x, z) \in C \times B \mid \exists y \in A. (x, y) \in g \wedge (y, z) \in f\}$. On objects, both products and coproducts are formed as in Set ; for morphisms we find $\coprod_{\nu < \alpha} f_\nu : \coprod_{\nu < \alpha} X_\nu \rightarrow \coprod_{\nu < \alpha} Y_\nu$ to be defined via $(\nu, x) \in (\coprod_{\nu < \alpha} f_\nu)$ iff $x \in f_\nu$; and $f_1 \times f_2 = \{((x_1, y_1), (x_2, y_2)) \mid \forall i \in \{0, 1\} (x_i, y_i) \in f_i\}$. It is a standard result that we find Rel to be a p-category with coproducts thus. We add the following in order to complete the setting:

Lemma 25. For any sets, A, B the relation \preceq defined via $P' \preceq P \Leftrightarrow \text{dom}(P') \subseteq \text{dom}(P) \wedge P \subseteq P'$ for $P, P' \subseteq A \times B$, is compatible with composition, products and coproducts in Rel , and meet-semilattice.

3.1 Computable many-one reductions (Type 1)

Let \mathcal{C}_1 to be the subcategory of Rel containing all partial computable functions $f : \subseteq \{0, 1\}^* \rightarrow \{0, 1\}^*$, where we identify a function with its graph. As the identity $\text{id}_{\{0, 1\}^*}$ is computable, and the composition of computable functions yields a computable function, this actually is a category. Moreover, the computable functions are closed under the formation of products and finite coproducts, and also contain all standard projections and injections, if we use the following definition:

Definition 26. For two multi-valued functions $f, g : \subseteq \{0, 1\}^* \rightrightarrows \{0, 1\}^*$, define $(f \coprod g), (f \oplus g) : \subseteq \{0, 1\}^* \rightrightarrows \{0, 1\}^*$ via $(f \coprod g)(0x) = 0f(x)$, $(f \coprod g)(1x) = 1g(x)$ and $(f \oplus g)(\langle x, y \rangle) = 0f(x) \cup 1g(y)$.

Thus, we find $(Rel|_{\{0, 1\}^*}, \mathcal{C}_1, \times, \preceq)$ to be a moce, and study $\mathfrak{C}_1 = \mathfrak{D}(Rel|_{\{0, 1\}^*}, \mathcal{C}_1, \times, \preceq)$. We give the special cases of the definitions and results from Section 2.2 below:

Definition 27 (special case of Definition 9). For two multi-valued functions $f, g : \subseteq \{0, 1\}^* \rightrightarrows \{0, 1\}^*$, define $f \leq_m g$, if there are computable functions $H, K : \subseteq \{0, 1\}^* \rightarrow \{0, 1\}^*$ with $H\langle x, y \rangle \in f(x)$ whenever $y \in g(K(x))$.

Corollary 28 (of Theorem 12). $(\mathfrak{C}_1, \oplus, \coprod)$ is a distributive lattice.

In \mathcal{C}_1 , there exists both an empty domain and final domains, namely the no-where defined multi-valued function $\emptyset \subseteq \{0, 1\}^* \times \{0, 1\}^*$ and any $\{(x, x)\} \subseteq \{0, 1\}^* \times \{0, 1\}^*$. The corresponding degrees shall be denoted by $0, 1 \in \mathcal{C}_1$.

Proposition 29. 1 is the least element in $\mathcal{C}_1 \setminus \{0\}$ and contains exactly those multi-valued functions admitting a computable choice function.

We do point out that decision problems cannot be considered as a special case of multi-valued functions in the straight-forward way, as our definition of many-one reductions allows modifications of the output; in particular, the characteristic function of a set is trivially equivalent to the characteristic function of its complement. However, many results proven for many-one reductions between search problems hold - with identical proofs - also for Turing reductions with the number of oracle queries limited to 1, which corresponds to the notion employed here.

For example, YATES' result regarding the existence of minimal pairs applies here as follows:

Proposition 30 ([38]). There are $\mathbf{a}, \mathbf{b} \in \mathcal{C}_1 \setminus \{0, 1\}$ with total representatives such that for any $\mathbf{c} \leq_m (\mathbf{a} \oplus \mathbf{b})$ that has a representative $f \in \mathbf{c}$ of the type $f : \{0, 1\}^* \rightarrow \{0, 1\}$, we find $\mathbf{c} = 1$.

However, the constraint on the type of some representative is crucial, as we will demonstrate below. Instrumental is the next technical lemma:

Lemma 31. There are Turing functionals Ψ, Φ , such that for all total multi-valued functions $f, g : \{0, 1\}^* \rightrightarrows \{0, 1\}^*$ and for any choice function I of $(f \oplus g)$, either Ψ^I is a choice function of f or Φ^I is a choice function of g .

Proof. On input x , Ψ will search for some y such that $I\langle x, y \rangle = 0z$. Once this is found, it will output z . On input y , Φ will search for some x such that $I\langle x, y \rangle = 1z$, and output z .

It is clear that if I is a choice function of $(f \oplus g)$, then $\Psi^I(x) \in f(x)$ and $\Phi^I(y) \in g(y)$ whenever the former values exist. It remains to show that for any suitable I , either $\Phi^I(x)$ exists for all x or $\Psi^I(y)$ exists for all y . Assume that $\Phi^I(x)$ does not exist for some x_0 . That means the search for some y with $I\langle x_0, y \rangle = 0z$ for some z was unsuccessful, hence, $I\langle x_0, y \rangle = 1z_y$ for all y . But this means that x_0 always is a solution to the search performed by Ψ^I , hence $\Psi^I(y)$ always exists. \square

Corollary 32. If $\mathbf{a}, \mathbf{b} \in \mathcal{C}_1$ have total representatives, then $\mathbf{a} \oplus \mathbf{b} = 1$ implies $\mathbf{a} = 1$ or $\mathbf{b} = 1$.

Proof. Let $f \in \mathbf{a}$ and $g \in \mathbf{b}$ be total. If $\mathbf{a} \oplus \mathbf{b} = 0$, then $f \oplus g$ has a computable choice function I . By Lemma 31, either Ψ^I is a (computable!) choice function of f , implying $\mathbf{a} = 1$; or Φ^I is a (computable!) choice function of g , implying $\mathbf{b} = 1$. \square

3.2 Polynomial-time many reductions (Type 1)

This time we consider the category \mathcal{P}_1 of polynomial-time computable partial functions $f : \subseteq \{0, 1\}^* \rightarrow \{0, 1\}^*$. This is closed under the products and coproducts given by Definition 26, hence, we can study $\mathfrak{P}_1 = \mathfrak{D}(\text{Rel}_{\{0, 1\}^*}, \mathcal{P}_1, \times, \preceq)$ in the usual way. It is worth noting that the same considerations apply to other usual resource-bounded reducibilities.

Definition 33. For two multi-valued functions $f, g : \subseteq \{0, 1\}^* \rightrightarrows \{0, 1\}^*$, define $f \leq_m^p g$, if there are polynomial-time computable functions $H, K : \subseteq \{0, 1\}^* \rightarrow \{0, 1\}^*$ with $H\langle x, y \rangle \in f(x)$ whenever $y \in g(K(x))$. Let \mathfrak{P}_1 denote the set of \leq_m -degrees of multi-valued functions of this type.

Corollary 34 (of Theorem 12). $(\mathfrak{P}_1, \oplus, \coprod)$ is a distributive lattice.

The empty and final domains of \mathcal{P}_1 are exactly those of \mathcal{C}_1 considered in Subsection 3.1; again we shall use 0 and 1 to denote the respective degrees. Also, we find the following counterpart to Proposition 29

Proposition 35. 1 is the least element in $\mathfrak{P}_1 \setminus \{0\}$ and contains exactly those multi-valued functions admitting a polynomial-time computable choice function.

As in Subsection 3.1, the many-one degrees of decision problems (Karp degrees [23]) are not a substructure of \mathcal{P}_1 in the natural way; however, many results proven about them also hold for polynomial-time Turing reductions with a single permitted oracle query, which do form a natural substructure.

Some results and their proofs can even be extended to include search problems; this shall be demonstrated for LADNER's density result [26, Theorem 2]. For this, note that two notions coinciding for single-valued functions differ for multi-valued functions, namely the existence of a computable choice function and the decidability of the graph. In accordance with Proposition 29, it makes sense to call those multi-valued functions satisfying the former condition *computable*. Additionally, the latter condition has the disadvantage of not being preserved downwards by many-one reductions. However, a decidable graph is the condition needed for the following theorem. Its proof closely resembles the one of [26, Theorem 2], which in turn was inspired by techniques from [4].

Theorem 36. Let $\mathbf{a}, \mathbf{b} \in \mathfrak{P}_1$ admit representatives with decidable graphs and satisfy $\mathbf{b} \not\leq_m^p \mathbf{a}$. Then there are $\mathbf{b}_0, \mathbf{b}_1 \in \mathfrak{P}_1$ with $\mathbf{b} = \mathbf{b}_0 \coprod \mathbf{b}_1$, $\mathbf{b}_i \not\leq_m^p \mathbf{a}$ and $\mathbf{b} \not\leq_m^p \mathbf{a} \coprod \mathbf{b}_i$ for both $i \in \{0, 1\}$.

Proof. Let $a \in \mathbf{a}$ and $b \in \mathbf{b}$ both have a decidable graph. The proof constructs a polynomial-time decidable set $D \subseteq \{0, 1\}^*$ such that representatives b_0, b_1 of $\mathbf{b}_0, \mathbf{b}_1$ fulfilling the given criteria are obtained as $b_0(x) = 0$, $b_1(x) = b(x)$ for $x \in D$, and $b_0(x) = b(x)$, $b_1(x) = 0$ otherwise. The set D will have the form $D = \{x \in \{0, 1\}^* \mid |x| \in D'\}$ for some $D' \subseteq \mathbb{N}$.

This definition of b_0, b_1 already ensures the first condition to be true. For the remain ones, a priority argument is employed. Using an enumeration R_e of polynomial-time many-one reductions, and the notation $R_e(f)$ for the multi-valued function arising from the application of the reduction procedure R_e to f , we obtain the following conditions to be satisfied:

$$(P_{4e}) \quad R_e(a \coprod b_0) \not\leq b$$

$$(P_{4e+1}) \quad R_e(a \coprod b_1) \not\leq b$$

$$(P_{4e+2}) \quad R_e(a) \not\leq b_1$$

$$(P_{4e+3}) \quad R_e(a) \not\leq b_0$$

The polynomial-time decision procedure for D now works in stages, such that on input x all stages $s \leq |x|$ are performed. A clock is employed to ensure that the computation for a stage s does not take longer than cs steps for some fixed constant c . In each stage the procedure searches exhaustively for a witness verifying the condition P_{n+1} , where n is the number of the last condition for that the search was successful. After the last stage, the procedure sets $x \in D$, iff the least number of an open condition is even.

In order to do this search, knowledge about a, b and D is needed. By assumption, the graphs of a and b are computable. The circularity in the definition of D is resolved by aborting the search in stage s , if any question $x \in D?$ for $|x| \geq s$ arises. For smaller inputs, the set D is already fixed at this stage. Finally, if the time for a stage runs out, the search is also aborted.

It remains to prove that for every condition a witness will eventually be found. Such a witness remains valid, hence, finding a witness proves truth of the condition. On the other hand, as the time available for the search increases unboundedly, if the condition is true, eventually a witness will be found. So let P_l be the condition with the least number that remains unsatisfied, and let s be the first stage in which a witness for P_l was sought.

Case $P_l = P_{4e}$ By construction, this implies $b_0(x) = 0$ for $|x| \geq s$. Hence, b_0 is polynomial-time computable and $\mathbf{a} \amalg \mathbf{b}_0 = \mathbf{a}$. But then $R_e(a \amalg b_0) \subseteq b$ implies $\mathbf{b} \leq_m \mathbf{a}$ in violation to the initial assumption.

Case $P_l = P_{4e+1}$ By construction, this implies $b_1(x) = 0$ for $|x| \geq s$. Hence, b_1 is polynomial-time computable and $\mathbf{a} \amalg \mathbf{b}_1 = \mathbf{a}$. But then $R_e(a \amalg b_1) \subseteq b$ implies $\mathbf{b} \leq_m \mathbf{a}$ in violation to the initial assumption.

Case $P_l = P_{4e+2}$ By construction, this implies $b_1(x) = b(x)$ for $|x| \geq s$. Hence, $\mathbf{b}_1 = \mathbf{b}$. But then $R_e(a) \subseteq b_1$ implies $\mathbf{b} \leq_m \mathbf{a}$ in violation to the initial assumption.

Case $P_l = P_{4e+3}$ By construction, this implies $b_0(x) = b(x)$ for $|x| \geq s$. Hence, $\mathbf{b}_0 = \mathbf{b}$. But then $R_e(a) \subseteq b_0$ implies $\mathbf{b} \leq_m \mathbf{a}$ in violation to the initial assumption.

As every case of the contrary assumption leads to a contradiction, the constructed set must fulfill the desired criteria. \square

Corollary 37. The degrees in \mathcal{P}_1 admitting decidable graphs are dense (in themselves).

A question that has received a lot of attention regarding (polynomial-time) many-one reductions between decision problems is about the existence and nature of minimal pairs. In terms of lattice theory⁸, this asks whether the degree 1 is meet-irreducible, and if not, what kind of pairs can satisfy $\mathbf{a} \oplus \mathbf{b} = 1$. Following the initial result by LADNER that minimal pairs for polynomial-time many-one reductions between decision problems exist [26], AMBOS-SPIES could prove that every computable super-polynomial degree is part of a minimal pair [2].

For search problems, however, the question remains open:

Open Question 38. Is $1 \in \mathcal{P}_1$ meet-irreducible?

The techniques used to construct a minimal pair in [26, 2] diagonalize against pairs of reductions R_e, R_f trying to prevent $R_e(a) = R_f(b)$ for the constructed representatives a, b . If the equality cannot be prevented, then one can prove that the resulting set is already polynomial-time decidable using a constant prefix of b , hence, polynomial-time decidable. However, for search problems any pair of reductions to a pair of search problems produces a search problem, namely $R_e(a) \cup R_f(b)$.

A non-computable minimal pair for Type-2 search problems was constructed in [21]. Here, the crucial part is the identifiability of hard and easy instances, which is not available in a Type-1 setting. The negative answer we obtained for computable many-one reductions in Subsection 3.1 relied on Lemma 31, which again cannot be transferred to the time-bounded case: There are polynomial-time decidable relations R such that neither R nor its inverse $\neg R^\dagger$ admit a polynomial-time choice function, even if $P = NP$ should hold⁹.

⁸Which are of course not applicable to the original setting.

⁹A counterexample can be constructed as follows. On input (x, y) , the decision procedure works in stages i , starting at $i = 1$. In stage $2i$, it tests $|x| \leq i \wedge |y| \leq 2^i$, deciding $(x, y) \in R$ if yes, and proceeding to the next stage otherwise. In stage $2i + 1$, it tests $|x| \leq 2^i \wedge |y| \leq i$, deciding $(x, y) \notin R$ if yes, and proceeding to the next stage otherwise.

3.3 (Continuous) Weihrauch-reductions

The study of Weihrauch-reductions, i.e. computable many-one reductions in the second order setting, for partial multi-valued functions on Baire space $(\mathbb{N}^{\mathbb{N}})$, formed the template for many results presented in the present paper, hence, we just summarize the respective statements here. The Weihrauch-degrees form a distributive lattice [34, 6], admit minimal pairs, form a Kleene-algebra and are neither a Heyting nor a Brouwer algebra [21].

As an example of many-reductions without a direct connection to computability or complexity, we point out that continuous Weihrauch-reducibility was actually introduced first. Again, all partial multi-valued functions on Baire space take the place of the super-category \mathcal{P} , while \mathcal{S} is given by the continuous partial functions on Baire space. Again, we obtain a distributive lattice [34, 6] and a Kleene-algebra, but minimal pairs no longer exist, instead the degrees form a Heyting algebra [21].

4 Non-standard examples

Besides the many-one reductions discussed in Section 3, there are also various interesting notions derived from other categories; justifying our general approach.

4.1 Parameterized Search Problems

While parameterized complexity theory [14, 18] is certainly well-established, parameterized search problems are only cursorily touched upon in [20]. However, the main ideas can readily be developed in our framework.

The crucial new element in parameterized complexity theory are parameterizations, which are usually defined to be polynomial-time computable functions $\kappa : \{0, 1\}^* \rightarrow \mathbb{N}$ [18, Definition 1.1]. We shall take the broader approach of considering any function $\kappa : \{0, 1\}^* \rightarrow \mathbb{N}$ as a parameterization initially. The parameterizations are the objects in our categories \mathcal{P}_{psp} , \mathcal{S}_{psp} . A \mathcal{P}_{psp} -morphism from κ_1 to κ_2 is any triple (κ_1, κ_2, R) where $R \subseteq \{0, 1\}^* \times \{0, 1\}^*$ is a partial multi-valued function. In particular, all hom-sets $\mathcal{P}_{psp}(\kappa_1, \kappa_2)$ are isomorphic in \mathcal{P}_{psp} .

The morphisms in \mathcal{S}_{psp} are those (κ_1, κ_2, R) where R is the graph of some function f such that there is an algorithm computing f with run-time bounded by $t(\kappa_1(w)) \cdot p(|w|)$ where t is some computable function and p some polynomial, and f furthermore satisfies $\kappa_2(f(w)) \leq F(\kappa_1(w))$ for some computable function $F : \mathbb{N} \rightarrow \mathbb{N}$. It is clear that \mathcal{S}_{psp} is closed under suitable composition, i.e. indeed a category.

Now we need products and coproducts of parameterizations. We can define these via $(\kappa_1 \times \kappa_2)((u, v)) = \max\{\kappa_1(u), \kappa_2(v)\}$ and $(\kappa_1 + \kappa_2)(iu) = \kappa_i(u)$. This allows us to define products and coproducts of morphisms in \mathcal{P}_{psp} by demanding $(\kappa_1, \kappa_2, R) \times (\kappa'_1, \kappa'_2, Q) = (\kappa_1 \times \kappa'_1, \kappa_2 \times \kappa'_2, R \times Q)$ and $(\kappa_1, \kappa_2, R) \coprod (\kappa'_1, \kappa'_2, Q) = (\kappa_1 + \kappa'_1, \kappa_2 + \kappa'_2, R \coprod Q)$. Straight-forward calculation verifies that these actually are products and coproducts (when amended with suitable projections and injections), and that \mathcal{S}_{psp} is closed under them.

Finally, we define $(\kappa_1, \kappa_2, R) \preceq (\kappa_1, \kappa_2, Q)$ to hold, iff $\text{dom}(R) \subseteq \text{dom}(Q) \wedge R \subseteq Q$ holds. We find that \preceq commutes with coproducts, products and composition, and that coproducts and products also commute. The infimum $\inf\{\iota_1\pi_1, \iota_2\pi_2\}$ always exists, hence, we find the parameterized search problems to form a distributive lattice as a corollary of Theorem 12, which we shall denote by \mathfrak{P}_{psp} . It is worth pointing out that the category of parameterized

search problems is a concrete poset enriched p-category over the category of search problems consider in Section 3.

At first it may seem surprising that a parameterized search problem has **two** parameterizations, not only one for the input, but also for the output. This does enable some nice structural results, for example the identity as a map from the parameterization κ_\perp to the parameterization κ_\top , where $\kappa_\perp(x) = 1$ and $\kappa_\top(x) = |x| + 1$, turns out to be complete for computable parameterized search problems.

However, for applications one might prefer to use a single parameterization to specify a search problem. In particular, this is a necessary step to consider parameterized approximation problems and parameterized counting problems as a special case of parameterized search problems. This can be achieved by fixing κ_\perp as the parameterization on the output side. One readily verifies $\kappa_\perp + \kappa_\perp = \kappa_\perp \times \kappa_\perp = \kappa_\perp$, hence, this restriction is compatible with the lattice operations. The definition of reductions then takes the form:

Definition 39. For simply parameterized search problems (κ_1, P) , (κ_2, Q) , let $P \leq_m Q$ hold, if there are functions F, G such that $x \mapsto F(\langle x, gG(x) \rangle)$ is a selector of P for any selector q of Q , additionally satisfying that $F(\langle x, y \rangle)$ is computable in time $f(\kappa_1(x)) \cdot p(|x| + |y|)$ and $G(x)$ is computable in time $f(\kappa_1(x)) \cdot p(|x|)$ for some computable function f and some polynomial p , and furthermore that $\kappa_2(G(x)) \leq g(\kappa_1(x))$ for some computable function g .

We find empty and initial domains in both \mathcal{P}_{psp} and \mathcal{S}_{psp} , namely the no-where defined multi-valued function with arbitrary parameterization, and all multi-valued functions $x \mapsto x$ for some constant $x \in \{0, 1\}^*$, again with arbitrary parameterization. As in Subsections 3.1, 3.2 we refer to the respective degrees by 0 and 1. We find again a counterpart to Propositions 29, 35:

Proposition 40. 1 is the least element in $\mathfrak{P}_{psp} \setminus \{0\}$ and contains exactly those parameterized multi-valued functions admitting a fixed parameter tractable choice function.

This it turn shows us that it is reasonable to consider any non- $\{0, 1\}$ degree of parameterized search problems as intractable. Comparison with the suggestion made before [20, Theorem 4.2] to regard a parameterized search problem as intractable, iff its tractability would imply tractability of a decision problem regarded as intractable invites the following question:

Open Question 41. Is there a non-fixed parameter tractable single-valued parameterized search problem with binary image below any non-fixed parameter tractable parameterized search problem?

4.2 Medvedev-reducibility

While Medvedev-reducibility [28] is not commonly regarded as a many-one reduction, we can nonetheless apply Theorem 12 in order to prove that it is a distributive lattice. For this, we choose $\mathcal{P}_{\mathfrak{M}}$ to be all computable partial multi-valued functions on Baire space, and $\mathcal{S}_{\mathfrak{M}}$ to be all computable partial functions on Baire space. Proceeding as in Subsection 3.3 for the remaining parts of a moce, we obtain just the **dual** of Medvedev reducibility, as demonstrated in [21].

References

- [1] E. Aarts, J. Korst & W. Michiels (2007): *Theoretical Aspects of Local Search*. Springer.
- [2] Klaus Ambos-Spies (1987): *Minimal Pairs for Polynomial Time Reducibilities*. In: *Computation Theory and Logic, LNCS 270*, Springer, pp. 1–13.

- [3] Mihir Bellare & Shafi Goldwasser (1994): *The complexity of decision versus search*. *SIAM Journal on Computing* 23, pp. 97–119.
- [4] A. Borodin, R. L. Constable & J. E. Hopcroft (1969): *Dense and non-dense families of complexity classes*. In: *10th Annual Symposium on Switching and Automata Theory*, pp. 7–19.
- [5] Vasco Brattka (2005): *Effective Borel measurability and reducibility of functions*. *Mathematical Logic Quarterly* 51(1), pp. 19–44.
- [6] Vasco Brattka & Guido Gherardi (2011): *Weihrauch Degrees, Omniscience Principles and Weak Computability*. *Bulletin of Symbolic Logic* 17, pp. 73–117. ArXiv:0905.4679.
- [7] Xi Chen & Xiaotie Deng (2005): *3-NASH is PPAD-Complete*. Technical Report 134, Electronic Colloquium on Computational Complexity.
- [8] Xi Chen & Xiaotie Deng (2005): *Settling the complexity of 2-player Nash-equilibrium*. Technical Report 134, Electronic Colloquium on Computational Complexity.
- [9] J.R.B. Cockett & P.J.W. Hofstra (2008): *Introduction to Turing categories*. *Annals of Pure and Applied Logic* 156(2-3), pp. 183–209.
- [10] Constantinos Daskalakis, Paul Goldberg & Christos Papadimitriou (2006): *The Complexity of Computing a Nash Equilibrium*. In: *38th ACM Symposium on Theory of Computing*, pp. 71–78.
- [11] Constantinos Daskalakis & Christos Papadimitriou (2005): *Three-Player Games are Hard*. Technical Report 139, Electronic Colloquium on Computational Complexity.
- [12] Constantinos Daskalakis & Christos Papadimitriou (2011): *Continuous Local Search*. In: *Proceedings of SODA*.
- [13] R. DiPaola & A. Heller (1987): *Dominical categories: Recursion theory without elements*. *Journal of Symbolic Logic* 52, pp. 594–635.
- [14] Rod Downey & Michael Fellows (1999): *Parameterized Complexity*. Springer.
- [15] Dominic Dumrauf & Tim Süß (2010): *On the Complexity of Local Search for Weighted Standard Set Problems*. In: Fernando Ferreira, Benedikt Löwe, Elvira Mayordomo & Luís Mendes Gomes, editors: *Programs, Proofs, Processes, Lecture Notes in Computer Science* 6158, Springer, pp. 132–140.
- [16] Kousha Etessami & Mihalis Yannakakis (2007): *On the Complexity of Nash Equilibria and Other Fixed Points (Extended Abstract)*. In: *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, pp. 113–123.
- [17] Alex Fabrikant, Christos Papadimitriou & Kunal Talwar (2004): *The complexity of pure Nash equilibria*. In: *STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, ACM, New York, NY, USA, pp. 604–612.
- [18] Jörg Flum & Martin Grohe (2006): *Parameterized Complexity Theory*. Springer.
- [19] William Gasarch & Georgia Martin (1999): *Bounded Queries in Recursion Theory*, *Progress in Computer Science & Applied Logic* 16. Birkhäuser.
- [20] Georg Gottlob (2005): *Computing cores for data exchange: new algorithms and practical solutions*. In: *PODS*, pp. 148–159.
- [21] Kojiro Higuchi & Arno Pauly. *The degree-structure of Weihrauch-reducibility*. arXiv 1101.0112. Available at <http://arxiv.org/abs/1101.0112>.
- [22] David S. Johnson, Christos H. Papadimitriou & Mihalis Yannakakis (1988): *How easy is local search?* *Journal of Computer and System Sciences* 37(1), pp. 79–100.
- [23] Richard M. Karp (1972): *Reducibility among combinatorial problems*. In: R. E. Miller & J.W. Thatcher, editors: *Complexity of Computer Computations*, Plenum, pp. 85–103.
- [24] Dexter Kozen (1990): *On Kleene Algebras and Closed Semirings*. In: *Proc. Math. Found. Comput. Sci., LNCS* 452, Springer, pp. 26–47.

- [25] Martin Kummer & Frank Stephan (1994): *Effective Search Problems*. *Mathematical Logic Quarterly* 40, pp. 224–236.
- [26] Richard E. Ladner (1975): *On the Structure of Polynomial Time Reducibility*. *Journal of the ACM* 22(1), pp. 155–171.
- [27] G. Longo & E. Moggi (1984): *Cartesian closed categories of enumerations and effective type structures*. In: Plotkin Khan, MacQueen, editor: *Symposium on semantics of data types*, LNCS 173, Springer.
- [28] Yuri T. Medvedev (1955): *Degrees of difficulty of mass problems*. *Doklady Akademii Nauk SSSR* 104, pp. 501–504. In Russian.
- [29] Valerie de Paiva (1989): *A Dialectica-like model of linear logic*. In: D. Pitt et al., editor: *Categories in Computer Science and Logic*, *Lecture Notes in Computer Science* 389, Springer.
- [30] Christos H. Papadimitriou (1994): *Computational Complexity*. Addison Wesley.
- [31] Christos H. Papadimitriou (1994): *On the complexity of the parity argument and other inefficient proofs of existence*. *Journal of Computer and Systems Science* 48(3), pp. 498–532.
- [32] Christos H. Papadimitriou (2007): *The complexity of finding Nash equilibria*. In: Noam Nisan, Tim Roughgarden, Éva Tardos & Vijay Vazirani, editors: *Algorithmic Game Theory*, Cambridge University Press, pp. 29–52.
- [33] Arno Pauly (2010): *How Incomputable is Finding Nash Equilibria?* *Journal of Universal Computer Science* 16(18), pp. 2686–2710.
- [34] Arno Pauly (2010): *On the (semi)lattices induced by continuous reducibilities*. *Mathematical Logic Quarterly* 56(5), pp. 488–502.
- [35] E. Robinson & G. Rosolini (1988): *Categories of partial maps*. *Information and Computation* 79(2), pp. 95 – 130.
- [36] Klaus Weihrauch (1992): *The degrees of discontinuity of some translators between representations of the real numbers*. *Informatik Berichte* 129, FernUniversität Hagen, Hagen.
- [37] Klaus Weihrauch (1992): *The TTE-interpretation of three hierarchies of omniscience principles*. *Informatik Berichte* 130, FernUniversität Hagen, Hagen.
- [38] C. E. M. Yates (1966): *A Minimal Pair of Recursively Enumerable Degrees*. *The Journal of Symbolic Logic* 31(2), pp. 159–168.